



Carnegie Mellon
Software Engineering Institute

Pittsburgh, PA 15213-3890

An Acquirer's Guide to Navigating Contractor Data

Jeannine Siviy
Wolf Goethert
Robert Ferguson

Software Engineering Measurement & Analysis Initiative

Sponsored by the U.S. Department of Defense
© 2004 by Carnegie Mellon University

Report Documentation Page			Form Approved OMB No. 0704-0188	
<p>Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p>				
1. REPORT DATE JAN 2004	2. REPORT TYPE	3. DATES COVERED 00-00-2004 to 00-00-2004		
4. TITLE AND SUBTITLE An Acquirer's Guide to Navigating Contractor Data		5a. CONTRACT NUMBER		
		5b. GRANT NUMBER		
		5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S)		5d. PROJECT NUMBER		
		5e. TASK NUMBER		
		5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Carnege Mellon University, Software Engineering Institute, Pittsburgh, PA, 15213		8. PERFORMING ORGANIZATION REPORT NUMBER		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)		
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)		
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited				
13. SUPPLEMENTARY NOTES				
14. ABSTRACT				
15. SUBJECT TERMS				
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 51
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified	19a. NAME OF RESPONSIBLE PERSON	



Trademarks and Service Marks

® Capability Maturity Model, Capability Maturity Modeling, Carnegie Mellon, CERT, CERT Coordination Center, CMM, and CMMI are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

SM Architecture Tradeoff Analysis Method; ATAM;CMM Integration; CURE; IDEAL; Interim Profile; OCTAVE; Operationally Critical Threat, Asset, and Vulnerability Evaluation; Personal Software Process; PSP; SCAMPI; SCAMPI Lead Assessor; SCAMPI Lead Appraiser; SCE; SEI; SEPG; Team Software Process; and TSP are service marks of Carnegie Mellon University.



Objectives

Establish a view of the acquirer and supplier/contractor roles and responsibilities.

Show how measurement and analysis skills for internal development can be recast for acquisition and contracting environments.

Address a prevalent question in the acquisition community:

- How can we conduct causal analysis when we no longer control the collection processes and/or data?



Outline

Acquisition roles & responsibilities

Measurement & analysis methods

Illustration

- background
- monitoring and oversight: progress analysis

Summary

References



Responsibility and Authority

Measuring project and product success is the same whether the project is internal or contracted:

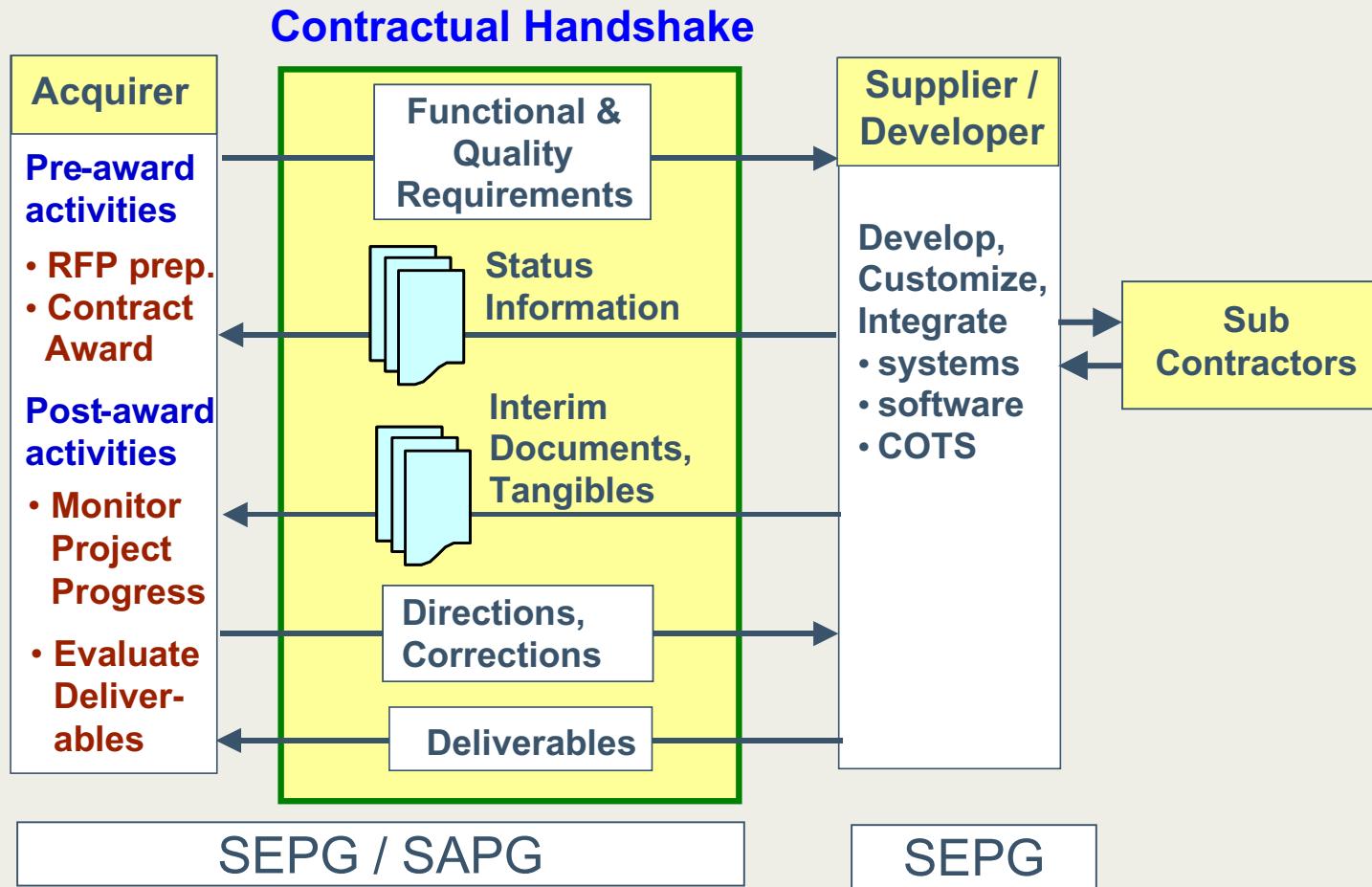
- on schedule
- at cost
- with required functionality
- without defects

The acquiring program manager's "circle of influence" and "circle of control" is different than the development project manager's.

- development project manager addresses project execution
- acquisition program manager executes new set of processes
- acquisition program manager should leverage development knowledge to manage the contract methodically, rationally, and knowledgeably

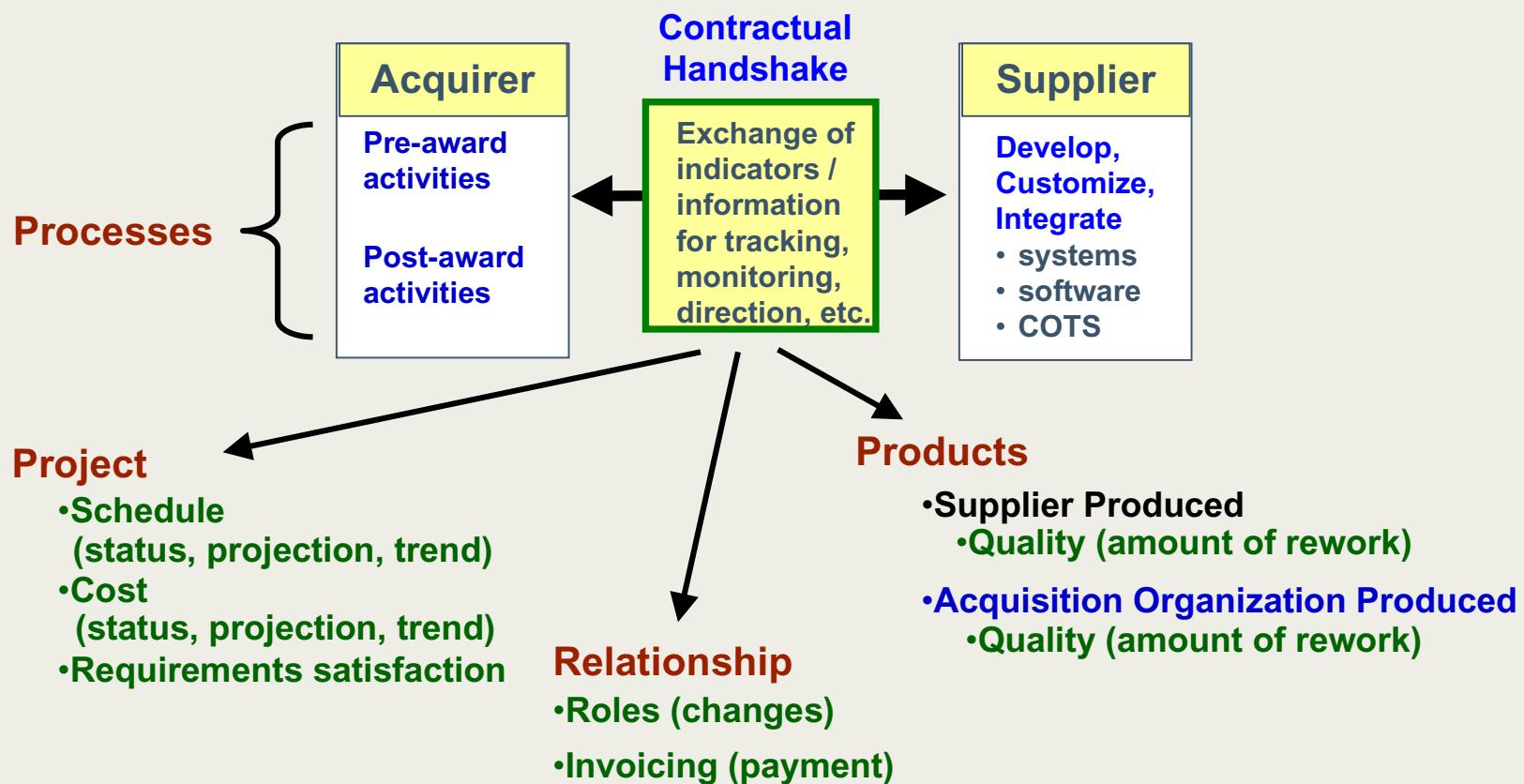


Roles and Information Exchange



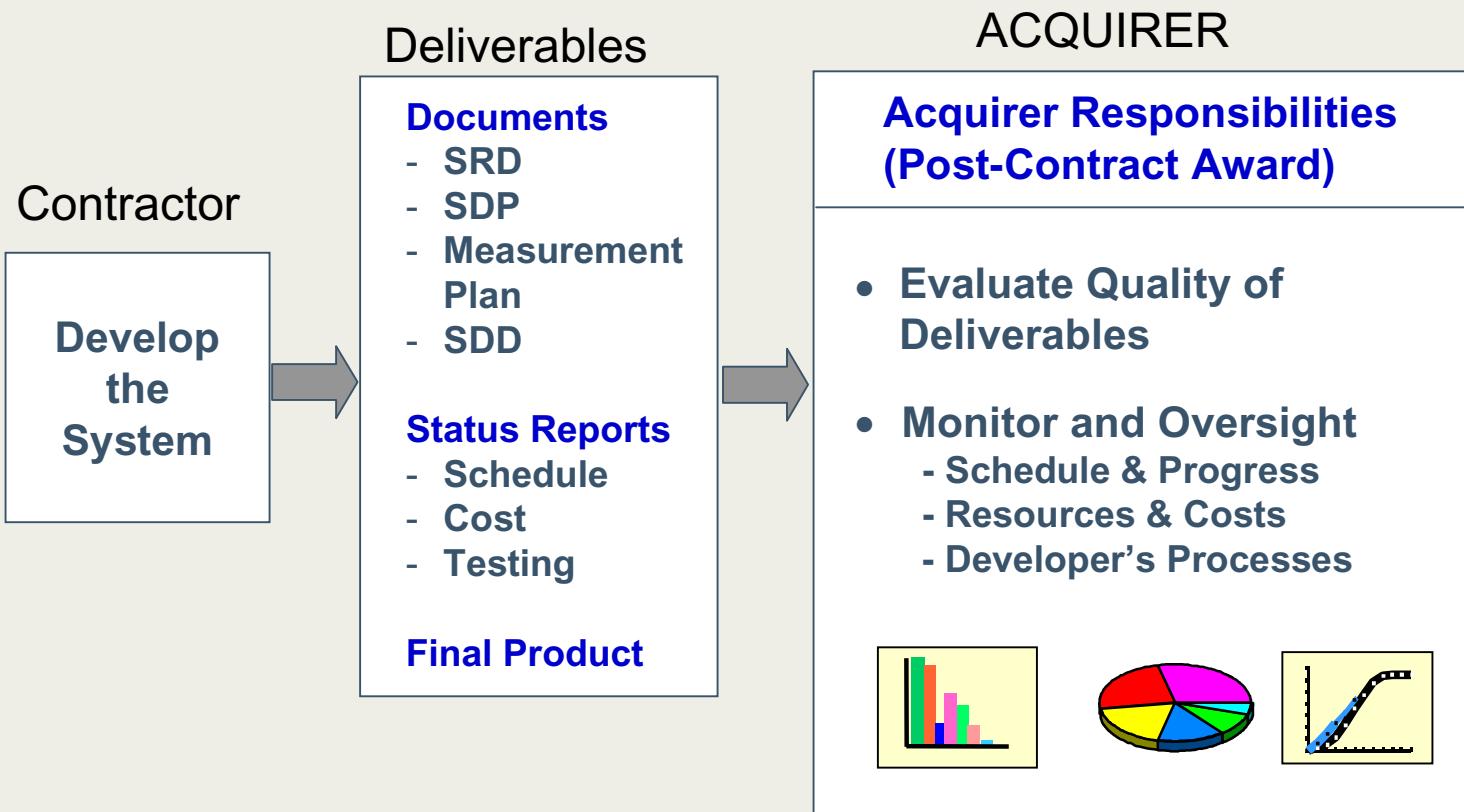


Measuring Project, Product, Process



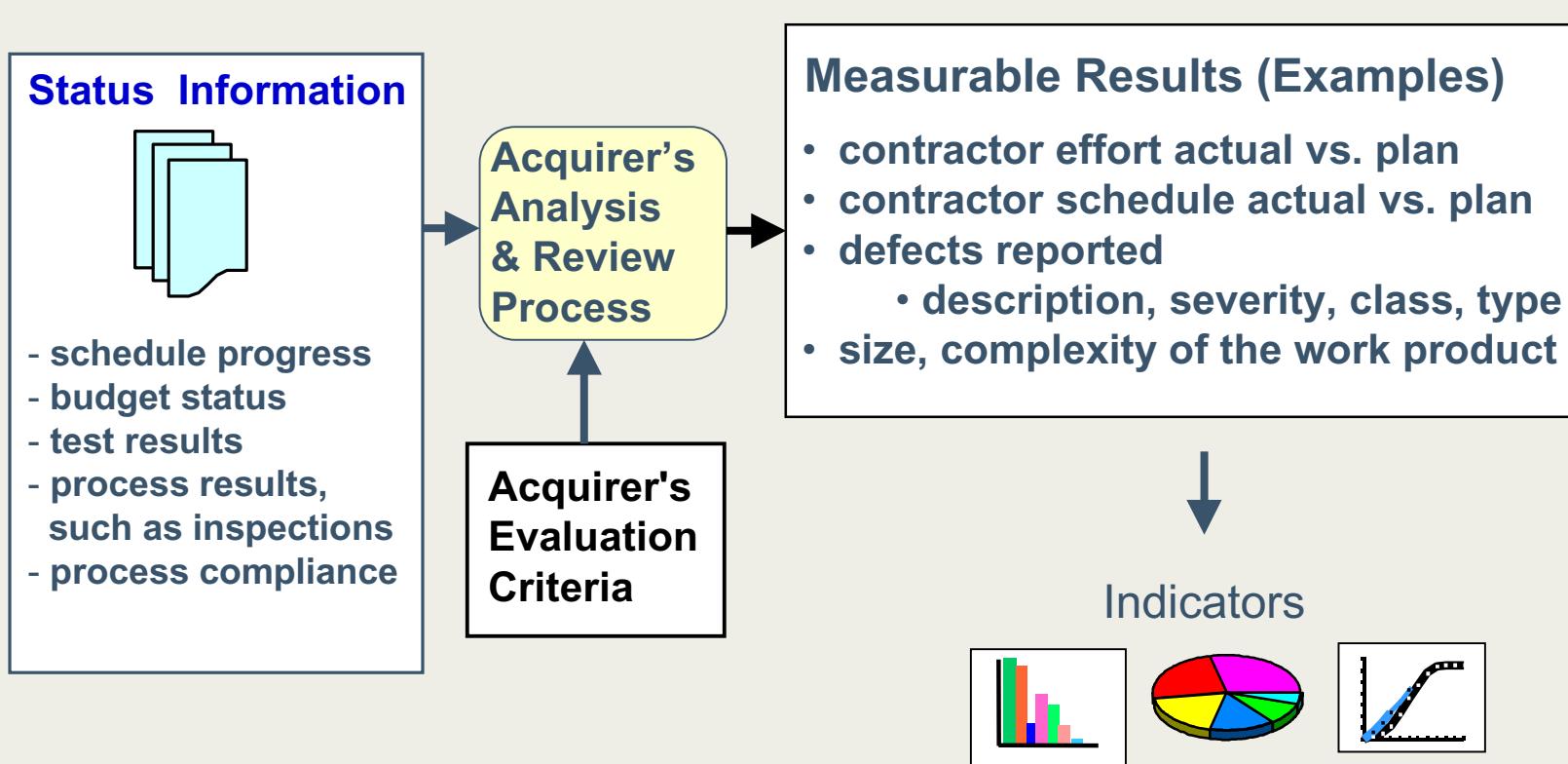


Responsibilities After Contract Award



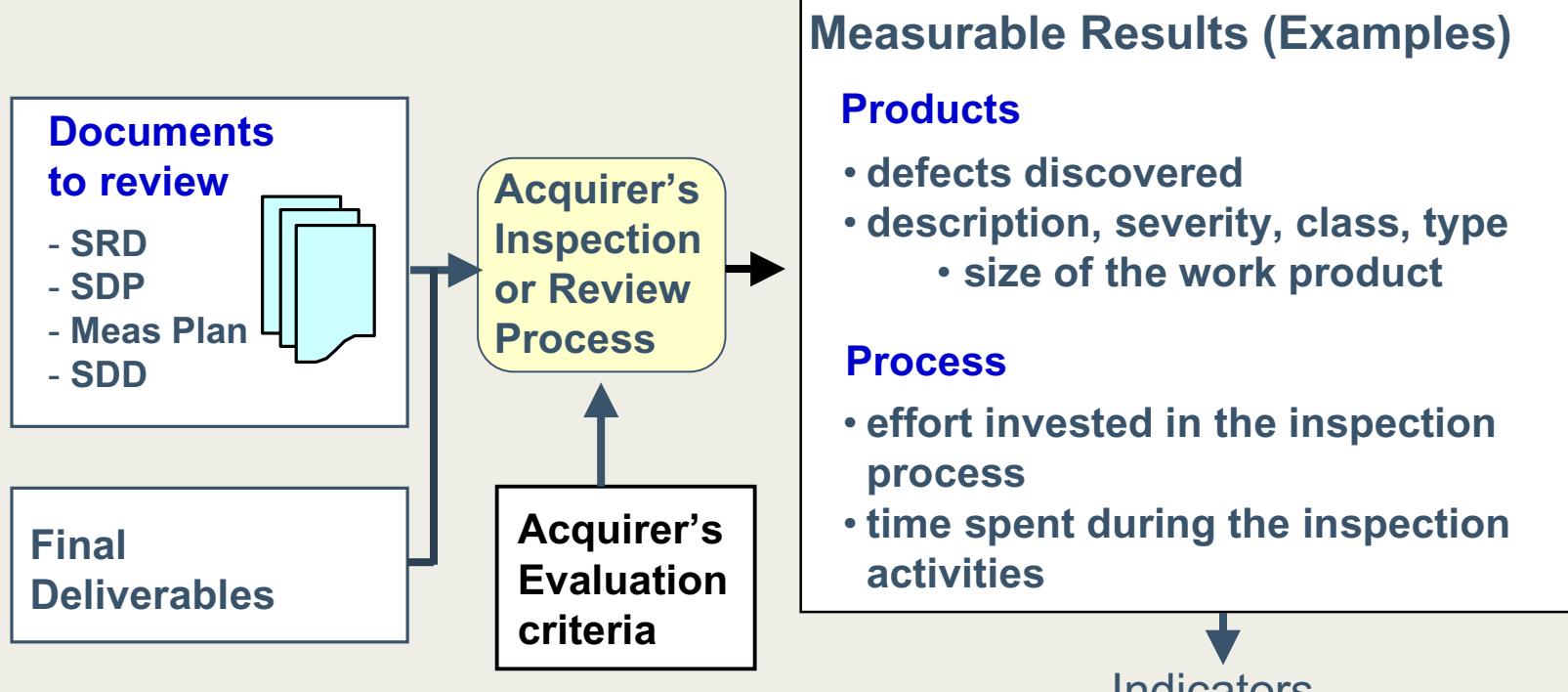


Monitoring & Oversight





Evaluating Quality of Deliverables





Outline

Acquisition roles & responsibilities

Measurement & analysis methods

Illustration

- background
- monitoring and oversight: progress analysis

Summary

References



Sources for Measures

Goal-Driven (Software) Measurement (GDM)

Goals → Questions → Indicators → Measures (GQIM)

USER DEFINES INDICATORS & MEASURES

Based On:

- what's needed to manage the User's goals
- decisions and decision criteria related to managing the user's goals

Practical Software & Systems Measurement

Common Issue Area → Measurement Category → Measures

PREDEFINED

PREDEFINED

PREDEFINED



Data Analysis Dynamics

Getting Started

- Identify the goals
- Black box process view
- Is the data right?
- Do I have the right data?

Decision point:

- If the data is not perfect, do I move forward or obtain better data?

Initial Evaluation

- What should the data look like?
- What does the data look like?
- Can I characterize the process, product, problem?

Decision point:

- Can I address my goals right now?
- Or is additional analysis necessary? at the same or deeper level of detail?
- Can I move forward?

Moving Forward

- Further evaluation
- Decompose data, process

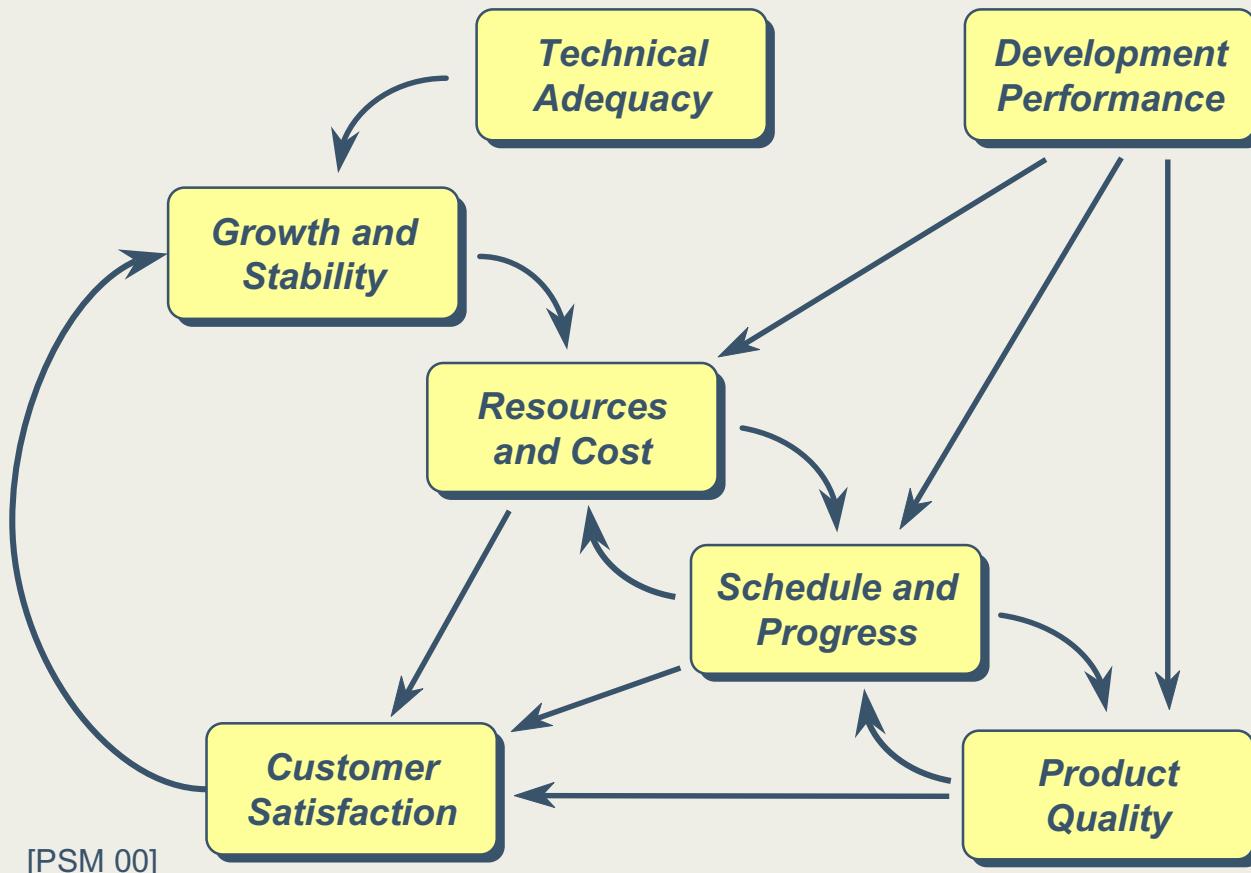
Decision point:

- Do I take action?
- What action do I take?

Repeat until root cause found, at target with desired variation



Performance Analysis Model





Performance Analysis Checklist 1

Single indicator issues:

- Do actual trends correspond to planned trends, such as progress, growth, and expenditures? How big is the variance?
- Does the variance appear to be gradually growing each month?
- Are actual values exceeding planned limits, such as open defects, changes, and resource utilization?

[PSM 00]



Performance Analysis Checklist ₂

Integrated indicator issues:

- Is the source of the problem evident?
 - Change in functionality, unplanned rework, etc.
- Are growing problems in one area a leading indicator of other problems later in the project?
 - Requirements creep impact on schedule
- Do multiple indicators lead to similar conclusions?
 - Lack of progress correlates with low staffing
- Does other project information contradict performance results?
 - Milestones being met but open defect counts are increasing

[PSM 00]



Outline

Acquisition roles & responsibilities

Measurement & analysis methods

Illustration

- background
- monitoring and oversight: progress analysis

Summary

References



Illustration

This illustration is based on an organization that is

- maintaining an existing product, a blend of COTS, and internally developed code
- pursuing the acquisition of a replacement product

Their acquisition includes two contracts:

- requirements development
- product design, code, and test

This illustration will focus on

- analyzing project execution data (Contract 2)



Monitoring & Oversight

Contract 2 has been awarded.

- supplier is developing the product in two builds

The contractor has just notified you that the project has both cost and schedule slippage.

What do you do?



Contractor Information

Contractor information:

- provides data per your contractual agreement
- also provides additional data if you ask*
- uses measurement data to help monitor development
- has defined processes and monitors compliance
- analyzes software trouble reports to identify process improvements
- average software process maturity

*Typically, If the RFP does not require the data, the contractor is not obligated to provide it. In this case study, the acquirer and contractor have a good working relationship and the contractor is willing to share data beyond what the RFP specifies.



Action: Management Review

Meet with development contractor to:

- Find out what is happening.
- Find out why it is happening.

What decisions/actions could be taken based upon the data presented?

- to correct the slippage
- to prevent further slippage

Postulate future consequences of these decisions

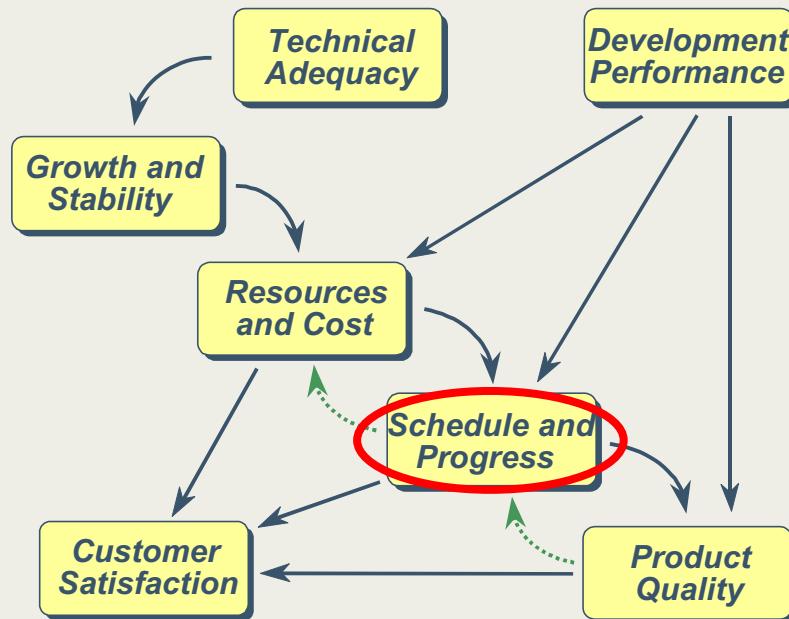
Identify actions that could have been taken earlier to prevent the slip



Performance Analysis Model

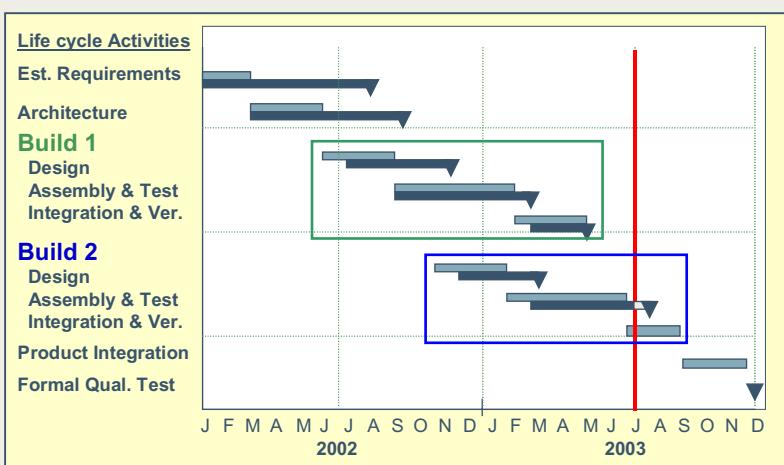
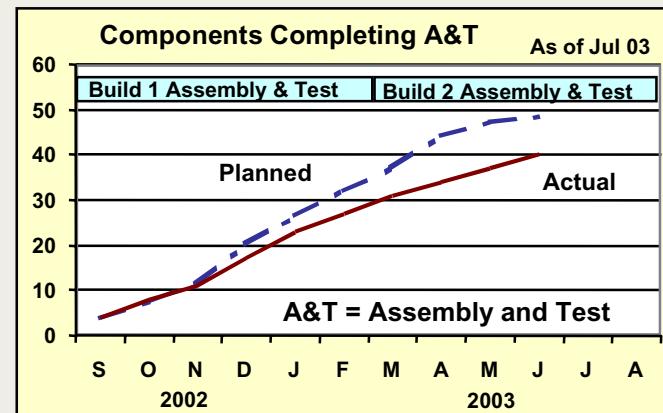
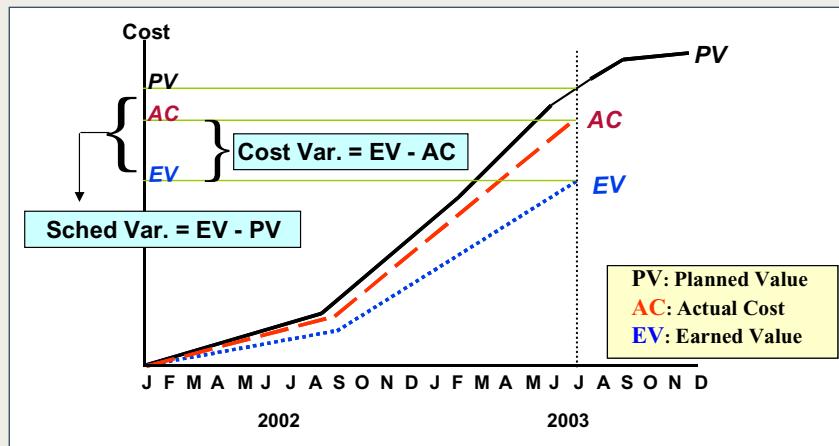
Use model to guide analysis.

- Step 1: Confirm Problem (Cost & Schedule Slippage)





Schedule & Progress Indicators



Tool tips:

The top two charts were made in Excel and manually manipulated.

The Gantt chart can be generated using any scheduling software.



What We Learned

From Schedule and Progress indicators

- cost and schedule slippage -- *EV chart*
- activities taking longer than planned -- *Gantt chart*
- assembly and test behind schedule -- *components completion chart*

What does this mean?

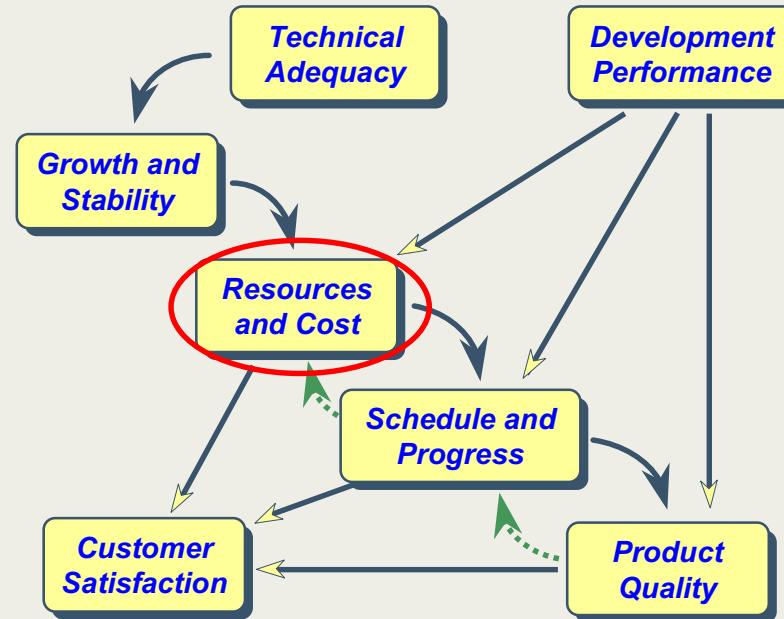
- confirms we have a problem



Resources and Cost Indicators

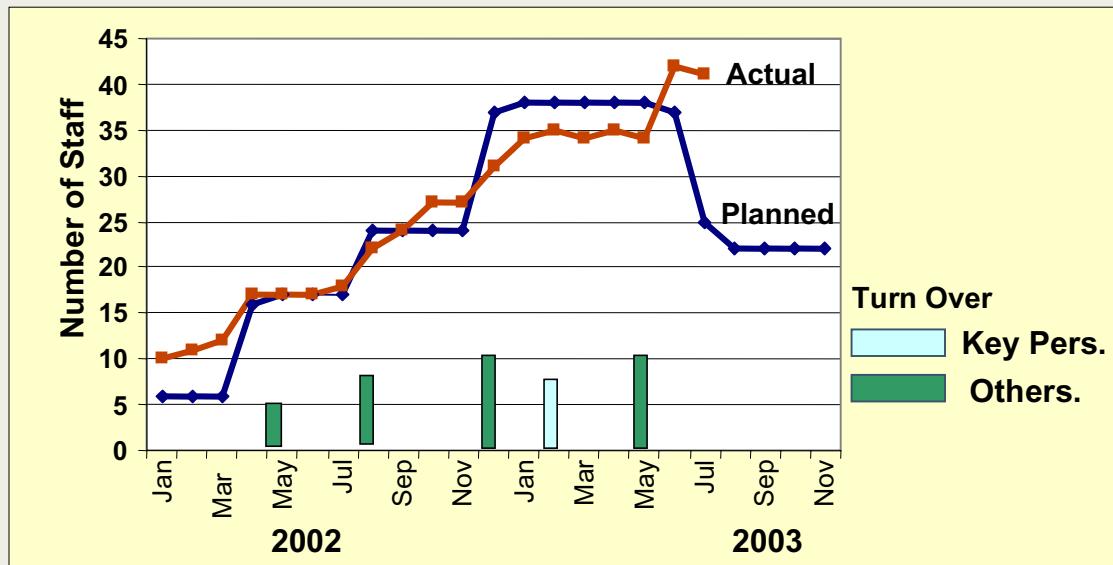
Analysis/Probing Questions

- Is the staff allocation contributing to the problem (too many, too few, wrong time frame)?
- What is rate of staff turnover?
- How does actual staff compare to planned staff allocation?





Resources and Cost Indicators



Prg	Plan	2	3	3	4	5	5	3	9	9	9	11	12	22	23	23	22	22	20	15
	Actual	3	3	4	5	5	5	5	6	9	9	9	11	11	15	18	19	19	20	30
Tester	Plan	3	3	3	3	3	3	1	1	1	3	3	3	2	5	5	7	8	8	5
	Actual	3	3	3	3	3	3	3	3	3	3	5	5	5	5	5	5	5	5	5

Tool tip: This chart was made in Excel and manually manipulated.



What We Learned

From Resources and Cost Indicators

- staffing did not follow planned level
 - too many at beginning of project
 - testers and programmers used to fill in for analysts and designers => high re-training costs
 - high turnover rate => training & getting up-to-speed costs

What does this mean?

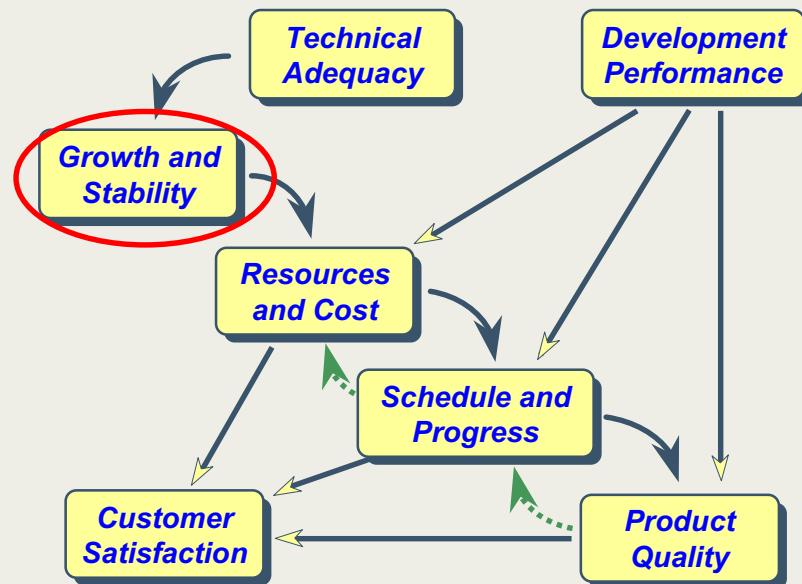
- cost overrun due partly to staffing problems



Growth and Stability Indicators

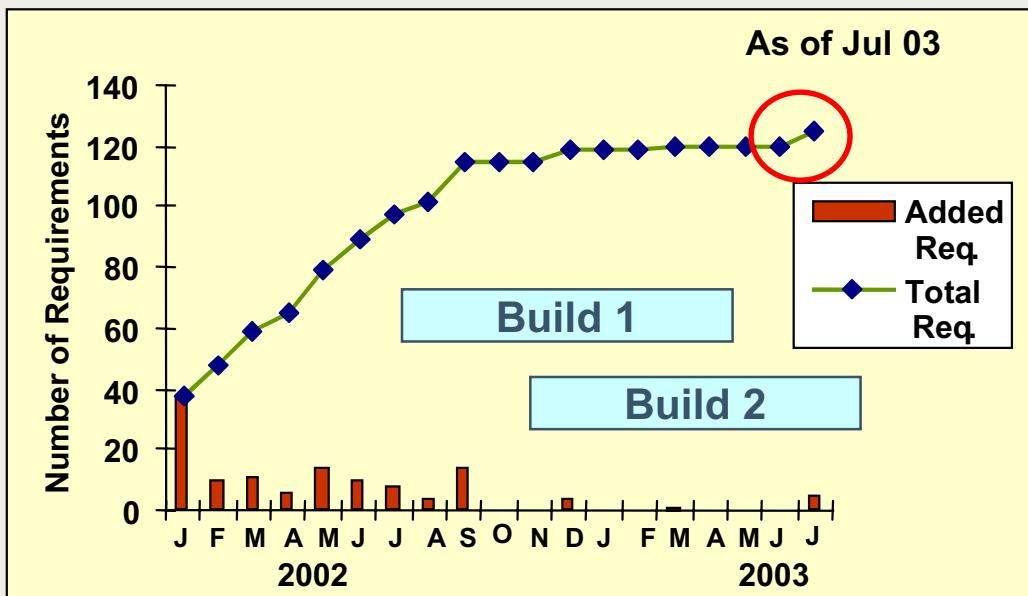
Analysis/Probing Questions

- Are the requirements stable?
- What is the code growth?
- Is functionality being transferred from build 1 to build 2? If so, how does this effect the delivery date?





Requirement Changes Information



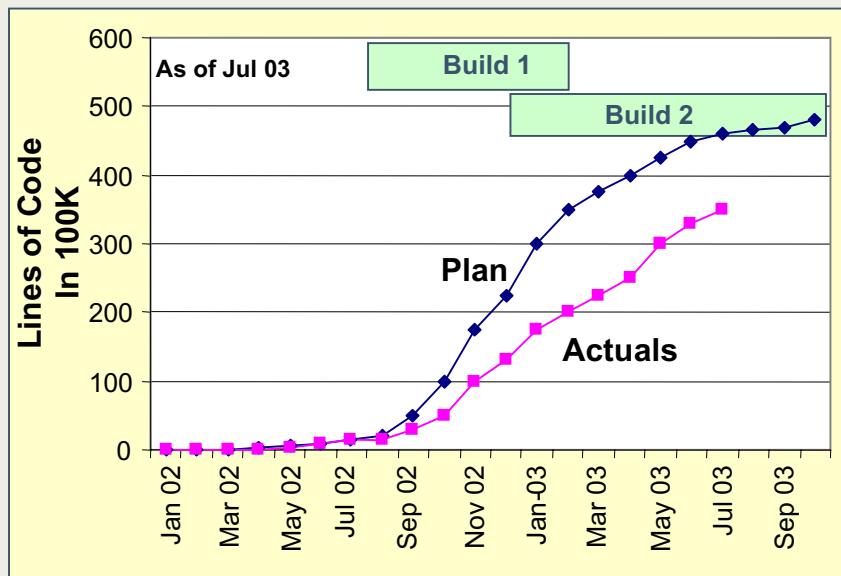
Tool tip: This chart can be generated in Excel followed by manual editing using the drawing toolbar

	2002								2003		
	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Dec	Mar	Jul
Req Changes	10	11	6	14	10	8	4	14	4	1	5
Complexity	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low
Resources (staff-days)	4	5	3	2	4	2	3	3	2	1	2

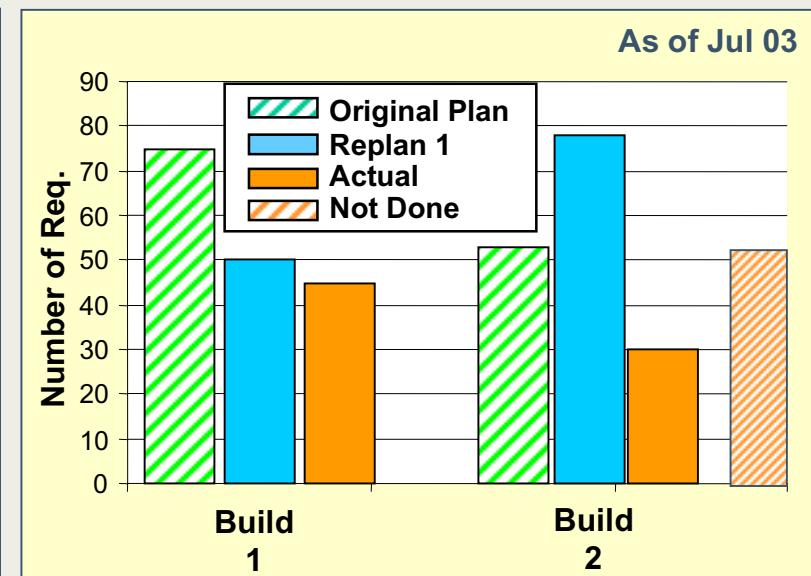


Growth and Stability Indicators

Size Growth



Requirements per Build



Tool tip: This chart was made in Excel and manually manipulated.

Contractor's Explanation:

- Functions deferred to later build because of unanticipated complexity



What We Learned

From Growth and Stability Indicators

- requirement changes are of low complexity but will have some ripple effect
- code production below planned value
- functionality being deferred from build 1 to build 2 attributed by contractor to unanticipated complexity

What does this mean?

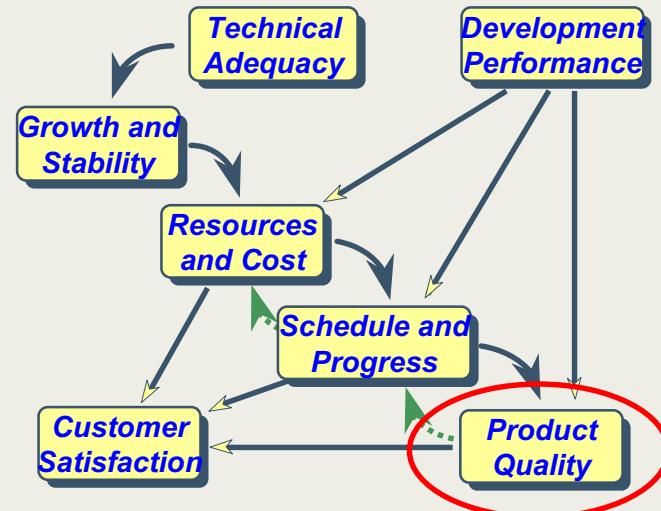
- expect further cost and schedule growth due to low code production and increased number of functions to be implemented in Build 2
- expect an impact on completion date due to functions deferred to Build 2
- expect the possibility of a “Build 3” proposal



Product Quality Indicators

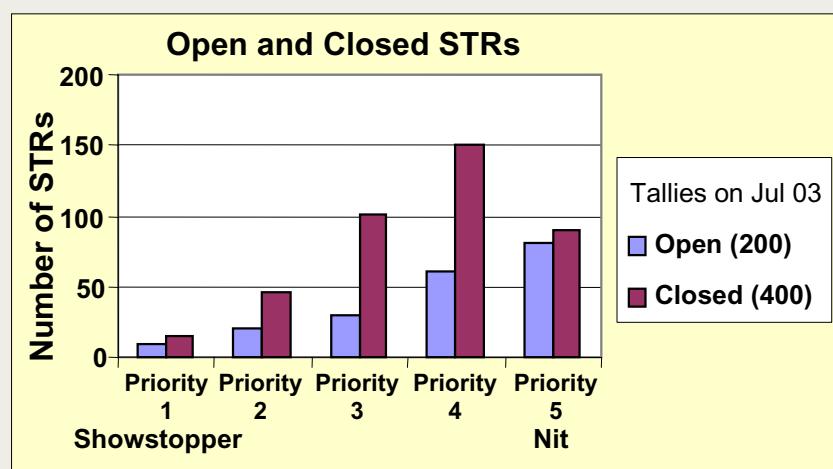
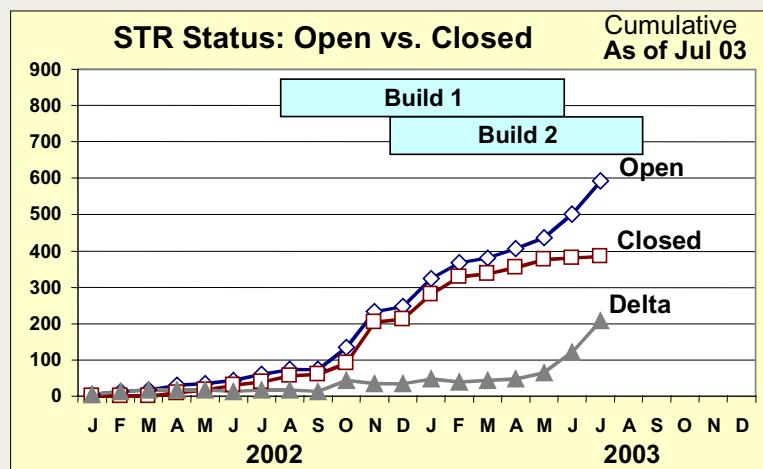
Analysis/Probing Questions

- Are the defined processes being followed?
- What is the rate of closure for trouble reports?
- What type of trouble reports are being detected? In what phase?





Product Quality Indicators

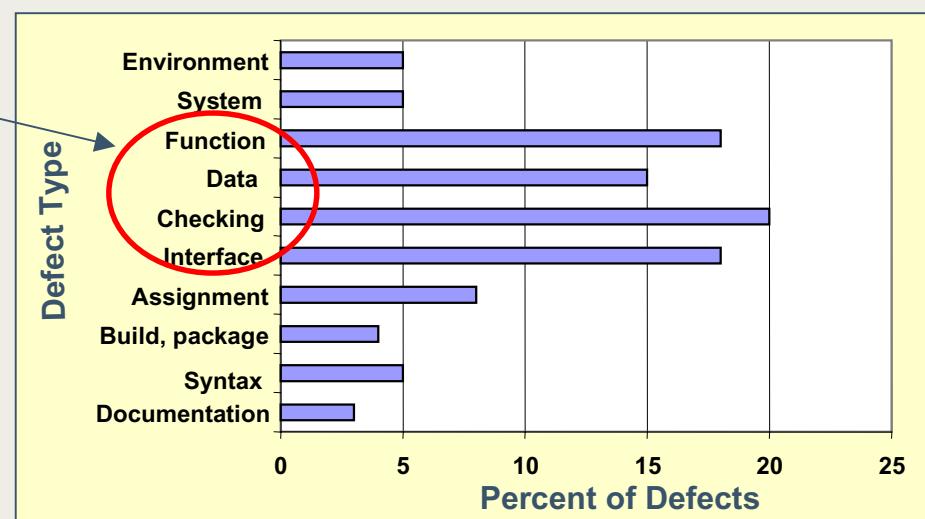


Tool tip: This chart was made in Excel and manually manipulated.



Classifying Trouble Report Defects

Types that code inspections would have been expected to catch





What We Learned

From Product Quality Indicators

- STRs being opened faster than they're being closed
- Code inspections should have found defect types

What does this mean?

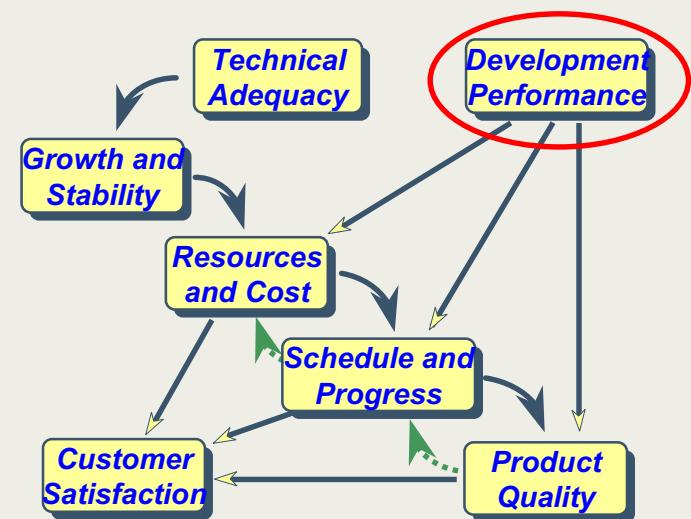
- Code inspection process allowed large number of defects to slip through.



Development Performance Indicators

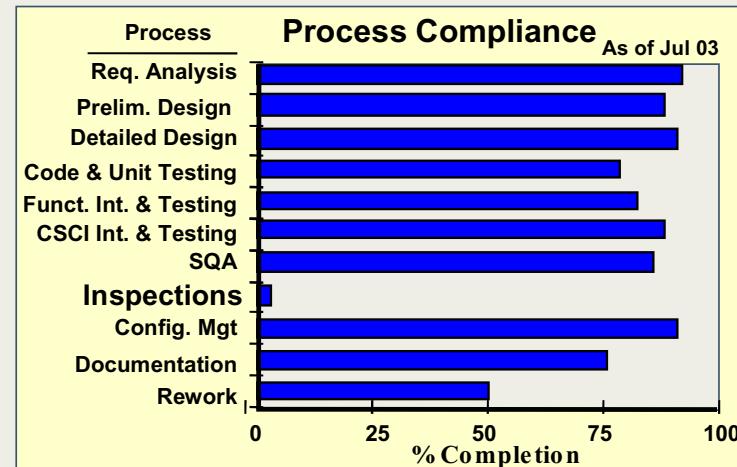
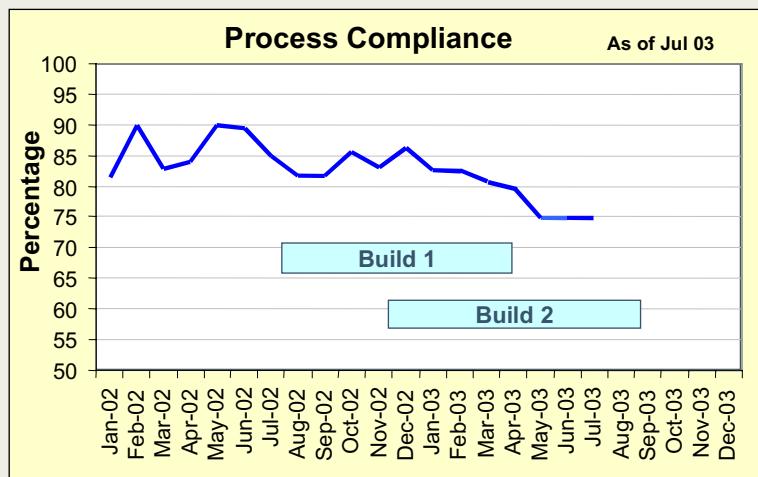
Analysis/Probing Questions

- Are the defined processes being followed?
- Are any defined processes being skipped?





Development Performance Indicators



Tool tip: This chart was made in Excel and manually manipulated.



What We Learned

From Development Performance Indicators

- adherence to defined process decreased over time
- stopped doing inspections

What does this mean?

- defects usually detected during code inspections allowed to slip through
- impact on cost and schedule due to rework



Reasons for Slippage

Staffing problems:

- too many at beginning of project
- below planned level during most of development
 - noting that productivity increased dramatically
- high turnover rate

Process compliance:

- stopped doing inspections
- allowed errors to leak to later phases

Requirements changes after Build 2 code and unit test

Conclusion:

- expect further cost and schedule growth due to low code production and increased number of functions to be implemented in Build 2



Possible Actions

Developer Actions

- replan based on current performance
- get staffing under control
 - verify the skills balance of resources
 - do not decrease staffing to conform to “planned” staffing, particularly if that would decrease the number of programmers
- restart inspections
 - code
 - test cases

Acquirer Decision Options

- use contract labor (additional costs)
- deliver smaller size - less functionality
- accept schedule slip



What if Data is Not Available?

Data may not be available because

- contractor does not collect this level of data
- contractor not required by contract to report it

Using process compliance data as an example, how might missing this data affect conclusions, actions, and project results?

- corrective action might have adjusted staffing
- it would not have addressed the skipped inspections which allowed errors to leak to later phases, resulting in increased cost and schedule

A possible action to infer process compliance

- could check data on results of code inspections (if data is specified on contract)

Lesson learned: specify in contract what type of data to be reported in status reports



Prevention

Visible indicators of underlying problems

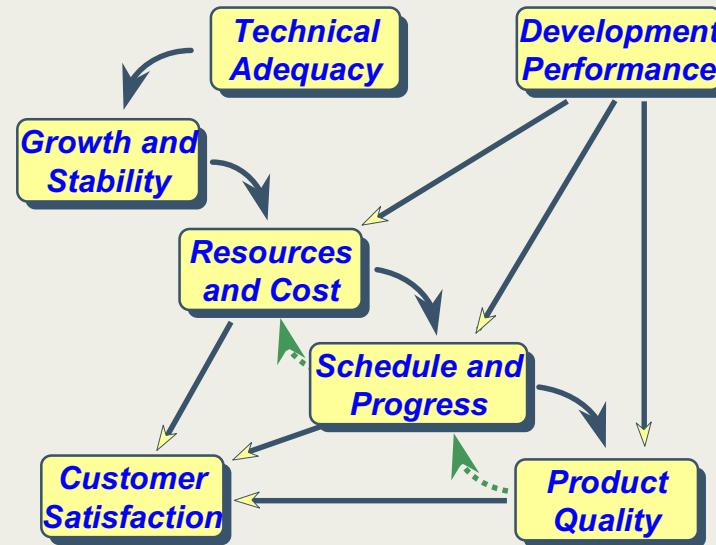
Cost & Schedule Slippage

Developers “corrective” actions

Functionality being deferred
Decreased process compliance
(skipped inspections)

Direct causes of problems

Unanticipated complexity
Inability to process STRs



Root causes of problems

Staffing issues
Requirement changes

The performance analysis model is a guide to root cause.
Understanding root cause leads to prevention.



Outline

Acquisition roles & responsibilities

Measurement & analysis methods

Illustration

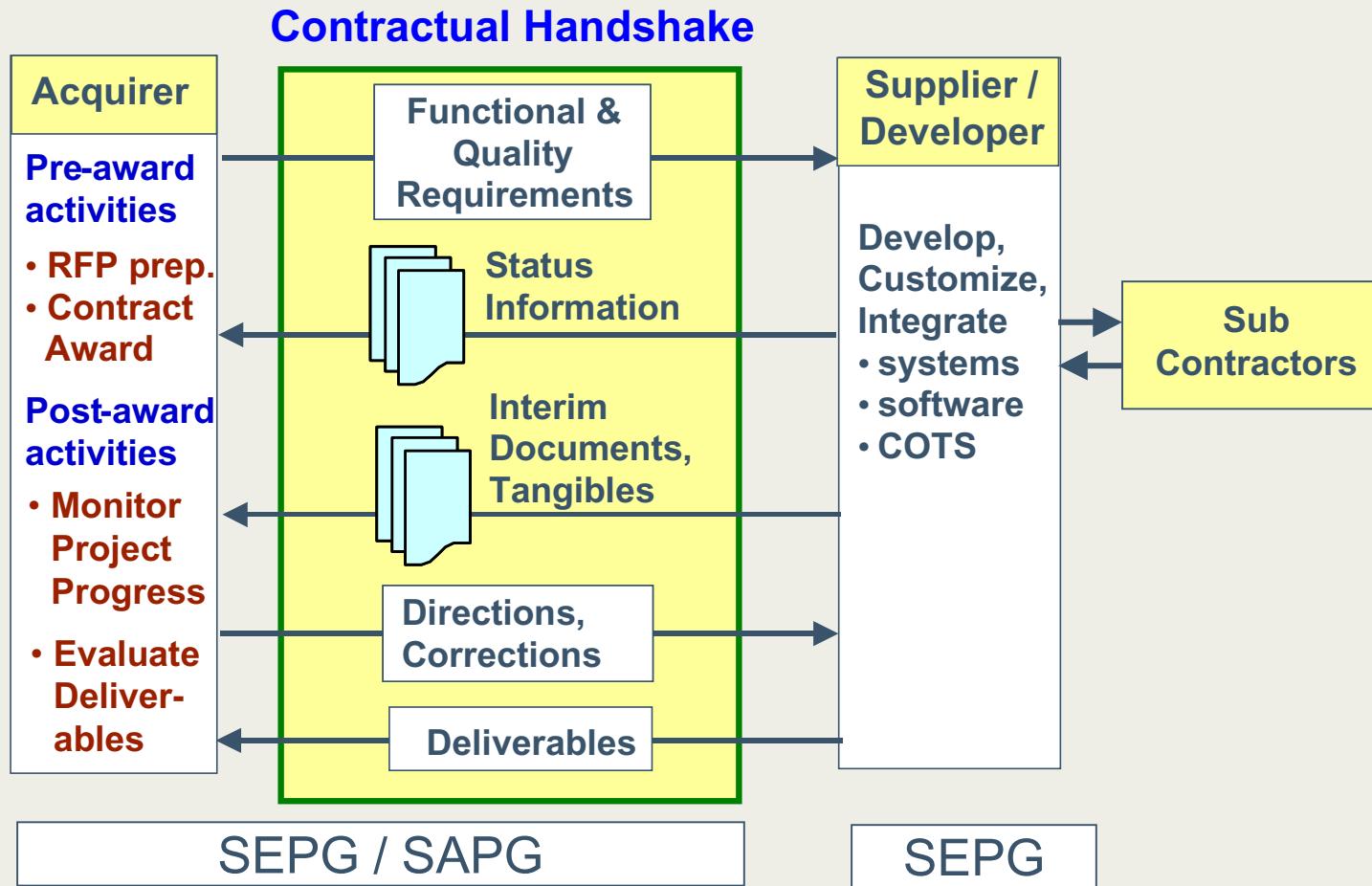
- background
- monitoring and oversight: progress analysis

Summary

References

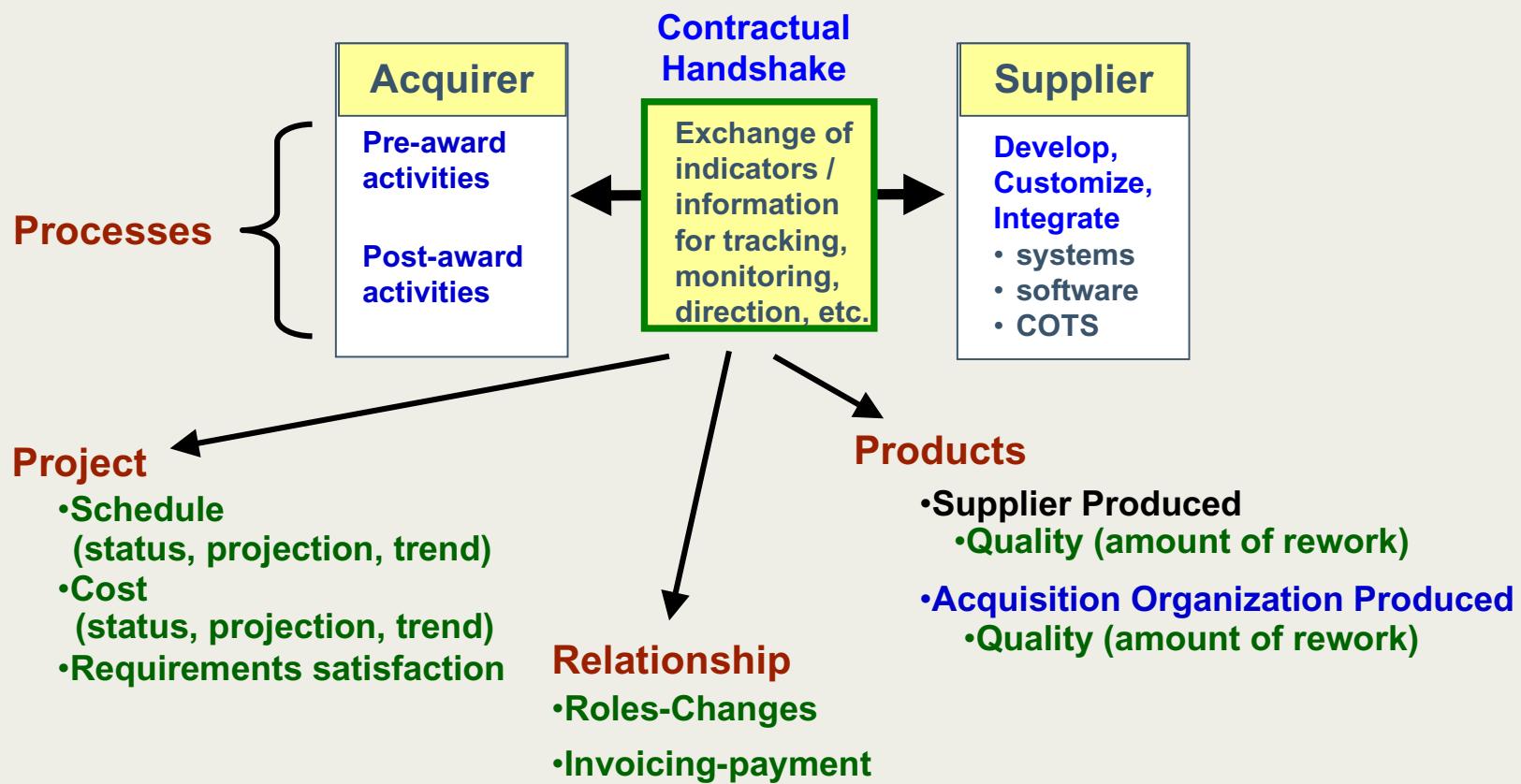


Roles and Information Exchange





Measuring Project, Product, Process





Summary

Key acquisition responsibilities (after contract award):

- monitoring and oversight
- inspecting, reviewing, and understanding documents and other work products

Post-contract award success depends on pre-contract award activities

- building measurement expectations into contracts
- establishing good partnerships and working relationships with contractors

Measures and indicators across landscape are interrelated

- use the Performance Analysis Model as your navigation guide
- always use multiple indicators

Measure products, processes, projects, relationships



Contact Information

Wolf Goethert
Software Engineering Institute
Measurement & Analysis Initiative
Email: wbg@sei.cmu.edu
412-268-3889

Jeannine Siviy
Software Engineering Institute
Measurement & Analysis Initiative
Email: jmsiviy@sei.cmu.edu
412-268-7994

Robert Ferguson
Software Engineering Institute
Measurement & Analysis Initiative
Email: rwf@sei.cmu.edu
412-268-9750



References 1

Note: URLs valid as of tutorial delivery date.

- [DAD 03] Siviy, Jeannine and William Florac, Data Analysis Dynamics, Half Day Tutorial Delivered at SEPG 2003, Boston, MA
- [GQIM 96] Goal-Driven Software Measurement--A Guidebook
<http://www.sei.cmu.edu/publications/documents/96.reports/96.hb.002.html>
- [PSM 00] Practical Software and Systems Measurement A Foundation for Objective Project Management, Guidebook, version 4.0b, Practical Software and Systems Measurement Support Center, U.S. TRACOM-ARDEC, AMSTA-AR-QA-A, Picatinny Arsenal, NJ, Website: www.psmsc.com, October 2000



Reading & Resources 1

Note: URLs valid as of tutorial delivery date.

Practical Software and Systems Measurement (PSM)

- reference for the Performance Analysis Model
- reference lists of measures to consider
- <http://www.psmsc.com>

Goal Driven Measurement (GDM) and Goal-Question-Indicator-Metric (GQIM)

- front end for selecting most relevant PSM measures
- used for developing context-specific indicators, particularly “success indicators”
- “Goal-Driven Software Measurement--A Guidebook”
<http://www.sei.cmu.edu/publications/documents/96.reports/96.hb.002.html>



Reading & Resources ₂

Note: URLs valid as of tutorial delivery date.

Defense Acquisition University (DAU) Deskbook

- <http://deskbook.dau.mil/jsp/default.jsp>
- provides information about regulatory references, mandatory and discretionary references by service branch, and several knowledge repositories

Guidelines for Successful Acquisition and Management of Software-Intensive Systems,

http://www.stsc.hill.af.mil/resources/tech_docs/index.html

Acquisition Centers of Excellence

- Air Force, for instance ESC Hanscom
 - <http://esc.hanscom.af.mil/ESC-BP/>
- Navy
 - <http://www.ace.navy.mil/public/html/>



Reading & Resources 3

Note: URLs valid as of tutorial delivery date.

Project Management Body of Knowledge (PMBOK®)

- proven, traditional project management practices and innovated, advanced practices with more limited use
- Project Management Institute Guide to the PMBOK contains the generally accepted subset of knowledge and practices that are applicable to most projects most of the time
 - http://www.pmi.org/info/PP_StandardsExcerpts.asp
 - http://www.pmi.org/info/PP_PMBOK2000Excerpts.asp